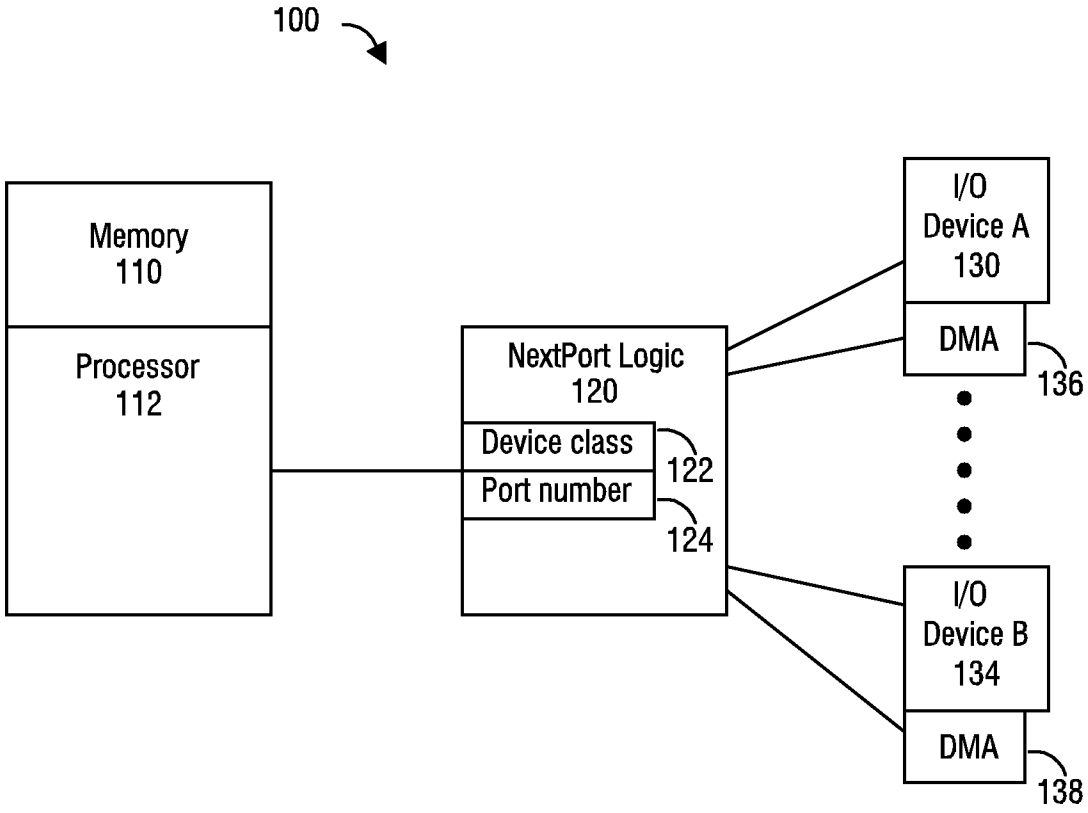

100
Memory
110
Processor
112
NextPort Logic
120
Device class
122
Port number
124
I/O
Device A
130
DMA
136
I/O
Device B
134
DMA
138

FIG. 1

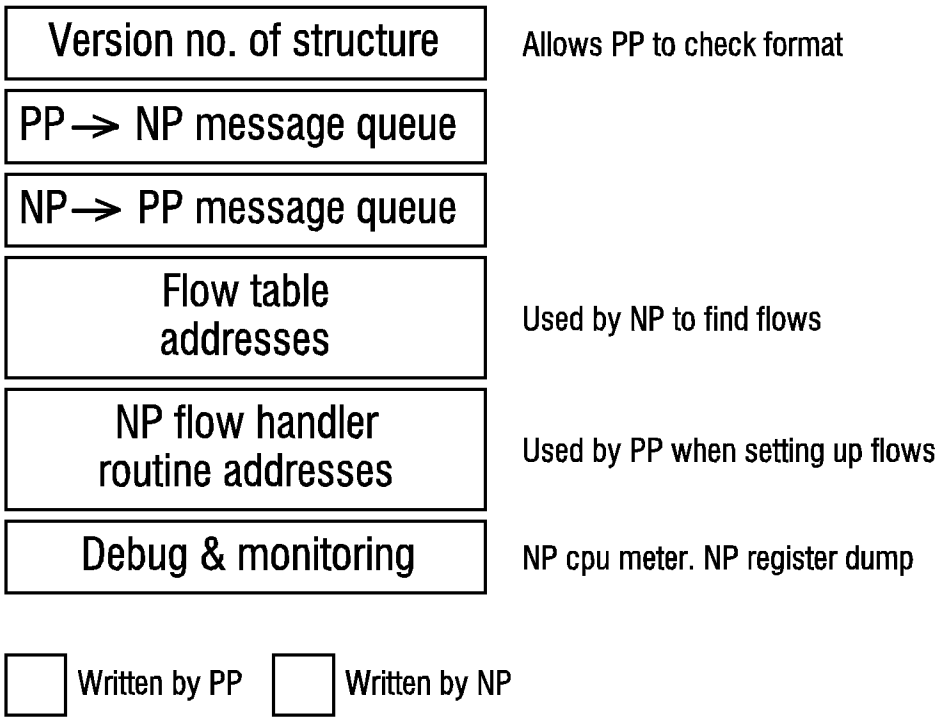
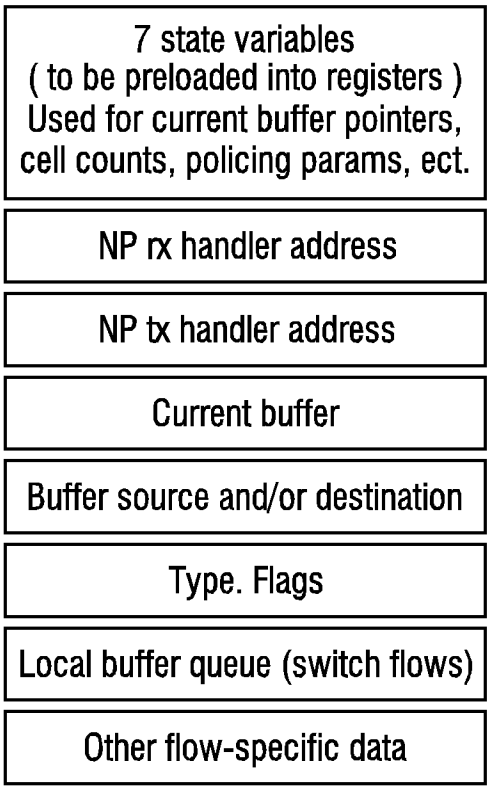

Version no. of structure
Allows PP to check format
PP → NP message queue
NP → PP message queue
Flow table addresses
Used by NP to find flows
NP flow handler routine addresses
Used by PP when setting up flows
Debug & monitoring
NP cpu meter. NP register dump
Written by PP
Written by NP

FIG. 4



First part has a similar format in all follows.

A flow is invoked by a single instruction:

- loads 8 or 9 registers
- jumps to handler routine

FIG. 5

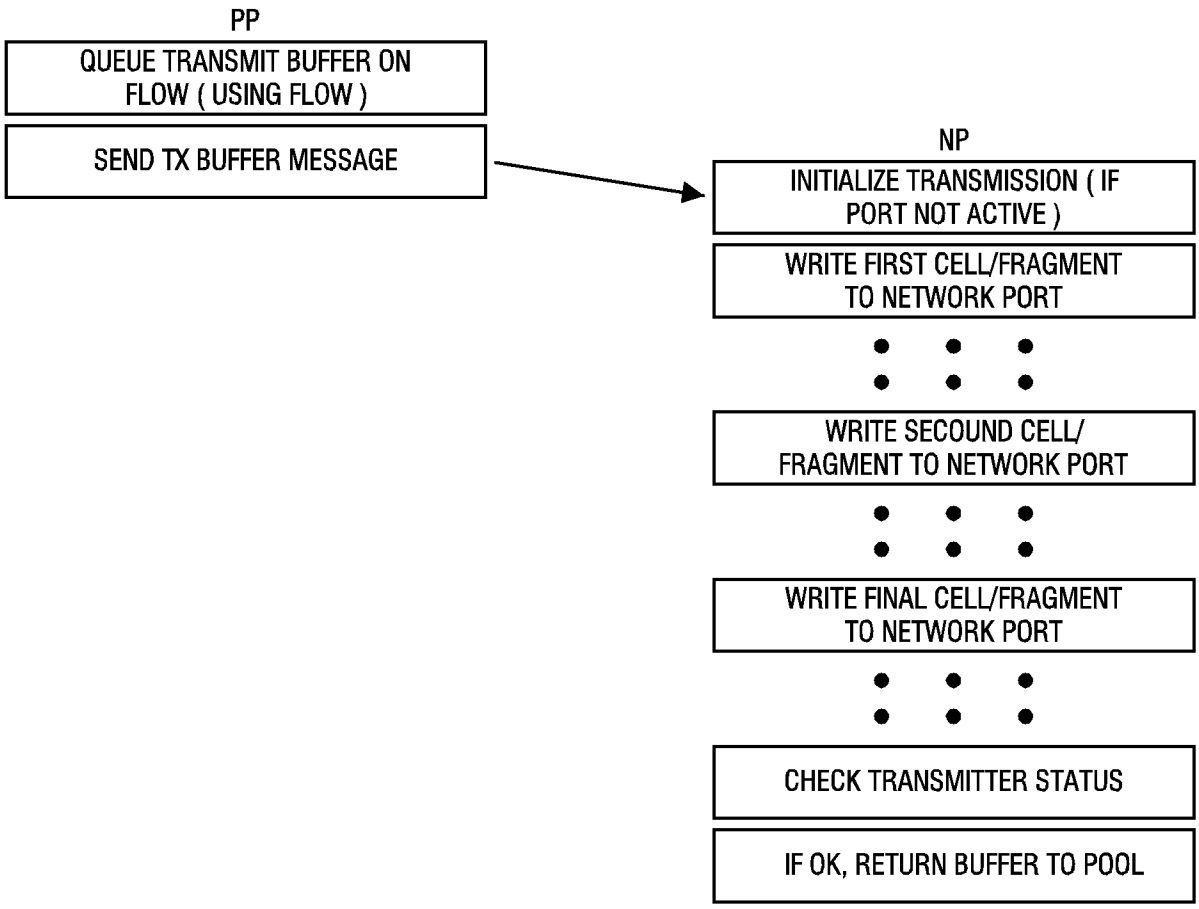

PP
QUEUE TRANSMIT BUFFER ON FLOW (USING FLOW)
SEND TX BUFFER MESSAGE
NP
INITIALIZE TRANSMISSION (IF PORT NOT ACTIVE)
WRITE FIRST CELL/FRAGMENT TO NETWORK PORT
WRITE SECOUND CELL/ FRAGMENT TO NETWORK PORT
WRITE FINAL CELL/FRAGMENT TO NETWORK PORT
CHECK TRANSMITTER STATUS
IF OK, RETURN BUFFER TO POOL

FIG. 6

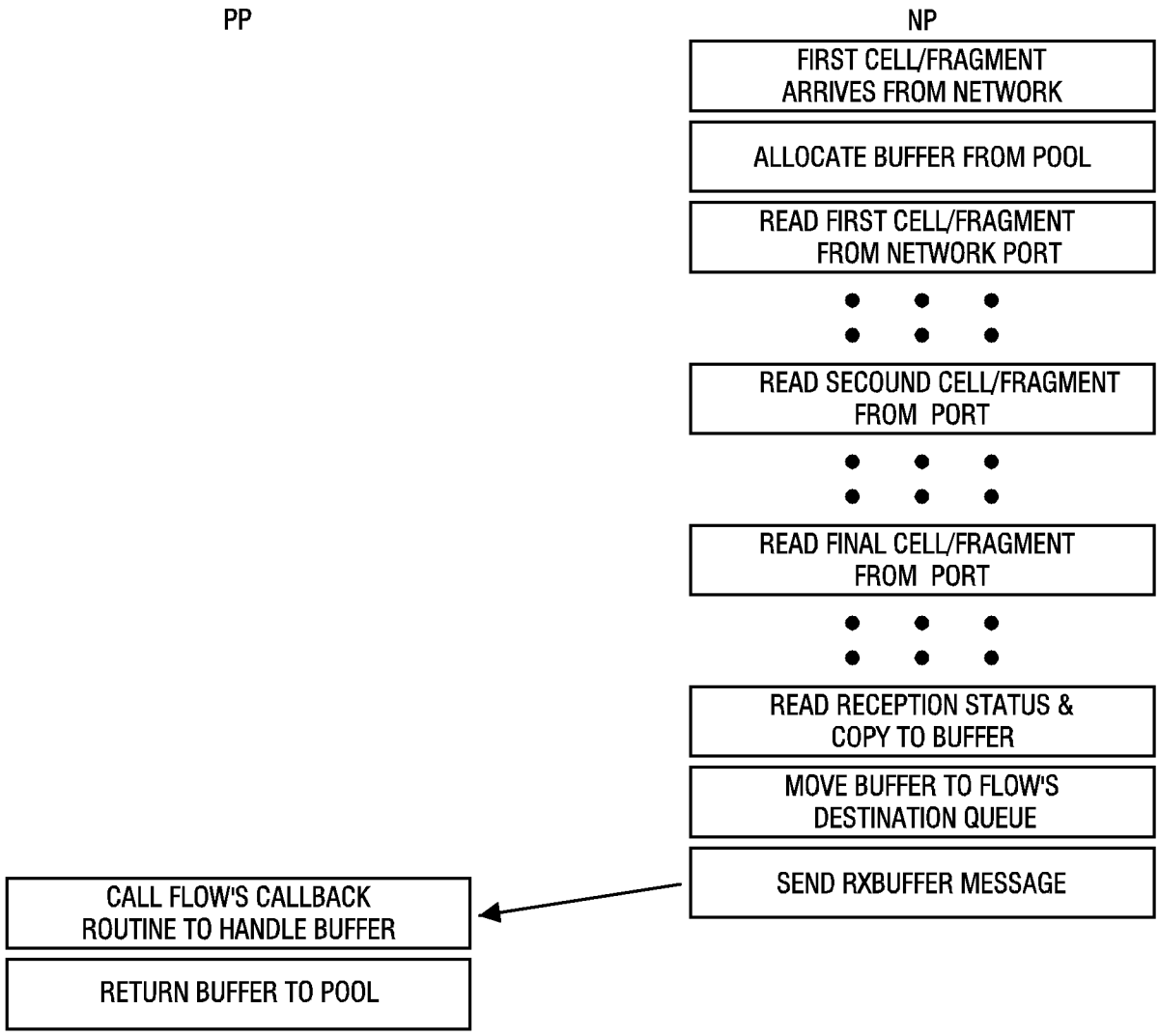

PP
NP
FIRST CELL/FRAGMENT ARRIVES FROM NETWORK
ALLOCATE BUFFER FROM POOL
READ FIRST CELL/FRAGMENT FROM NETWORK PORT
READ SECOUND CELL/FRAGMENT FROM PORT
READ FINAL CELL/FRAGMENT FROM PORT
READ RECEPTION STATUS & COPY TO BUFFER
MOVE BUFFER TO FLOW'S DESTINATION QUEUE
SEND RXBUFFER MESSAGE
CALL FLOW'S CALLBACK ROUTINE TO HANDLE BUFFER
RETURN BUFFER TO POOL

FIG. 7

OCTETS field in TX_ALIGN field in	Least significant 2 bite of DMA adress	Keep _ALIGN flag	TEXT_ALIGN register word at start	Next word from memory	TEXT_ALIGN register word after first memory cycle	Word written to FIFO
XX	00	0	XXXX . XXXX	pqrs . vwyz	XXXX . XX00	pqrs . vwyz
XX	01	0	XXXX . XXXX	pqrs . vwyz	pqrs . vw01	No write
XX	10	0	XXXX . XXXX	pqrs . XXXX	pqrs . vw02	No write
XX	11	0	XXXX . XXXX	pqXX . XXXX	pqXX . vw03	No write
00	00	1	XXXX . XX00	pqrs . vwyz	XXXX . XX00	pqrs . vwyz
00	01	1	XXXX . XX00	pqrs . vwXX	pqrs . vw01	No write
00	10	1	XXXX . XX00	pqrs . XXXX	pqrs . vw02	No write
00	11	1	XXXX . XX00	pqXX . XXXX	pqXX . vw03	No write
01	00	1	ghij . klo1	pqrs . vwyz	pqrs . vw01	yagh . ijk1
01	01	1	ghij . klo1	pqrs . vwXX	pqrs . vw02	vwgh . ijk1
01	10	1	ghij . klo1	pqrs . XXXX	pqXX . vw03	ragh . ijk1
01	11	1	ghij . klo1	pqXX . XXXX	XXXX . XX00	pqgh . ijk1
10	00	1	ghij . XX02	pqrs . vwyz	pqrs . vw02	vwyz . ghij
10	01	1	ghij . XX02	pqrs . vwXX	pqXX . vw03	rsvw . ghij
10	10	1	ghij . XX02	pqrs . XXXX	XXXX . XX00	pqrs . ghij
10	11	1	ghij . XX02	pqXX . XXXX	pqgh . ij01	No write
11	00	1	ghXX . XX03	pqrs . vwyz	pqXX . vw03	rsvw . yzgh
11	01	1	ghXX . XX03	pqrs . vwXX	XXXX . XX00	pqrs . vwgh
11	10	1	ghXX . XX03	pqrs . XXXX	pqrs . gh01	No write
11	11	1	ghXX . XX03	pqXX . XXXX	pqgh . XX02	No write

FIG. 10

OCTETS field in TX_ALIGN register	TEXT_ALIGN register word at start	word writting to FIFO register	FIFO register written	TEXT_ALIGN after FIFO register write	Word written to FIFO
00	XXXX . XX00	pqrs . vwyz	TX . FIF00	XXXX . XX00	pqrs . vwyz
00	XXXX . XX00	XXrs . vwyz	TX . FIF01	rsvw . yz01	No write
00	XXXX . XX00	XXXX . vwyz	TX . FIF02	vwzy . XX02	No write
00	XXXX . XX00	XXXX . XXyz	TX . FIF03	yzXX . XX03	No write
01	ghij . klo1	pqrs . vwyz	TX . FIF00	pqrs . vw01	yzgh . ijk1
01	ghij . klo1	XXrs . vwyz	TX . FIF01	rsvw . XX02	yzgh . ijk1
01	ghij . klo1	XXXX . vwyz	TX . FIF02	vwXX . XX03	yzgh . ijk1
01	ghij . klo1	XXXX . XXyz	TX . FIF03	XXXX . XX00	yzgh . ijk1
10	ghij . XX02	pqrs . vwyz	TX . FIF00	pqrs . XX02	vwyz . ijk1
10	ghij . XX02	XXrs . vwyz	TX . FIF01	rsXX . XX03	vwyz . ijk1
10	ghij . XX02	XXXX . vwyz	TX . FIF02	XXXX . XX00	vwyz . ijk1
10	ghij . XX02	XXXX . XXyz	TX . FIF03	pqgh . ij01	No write
11	ghXX . XX03	pqrs . vwyz	TX . FIF00	pqXX . XX03	rsvw . yzgh
11	ghXX . XX03	XXrs . vwyz	TX . FIF01	XXXX . XX00	rsvw . yxgh
11	ghXX . XX03	XXXX . vwyz	TX . FIF02	vwyz . gh01	No write
11	ghXX . XX03	XXXX . XXyz	TX . FIF03	yzgh . XX02	No write

FIG. 11

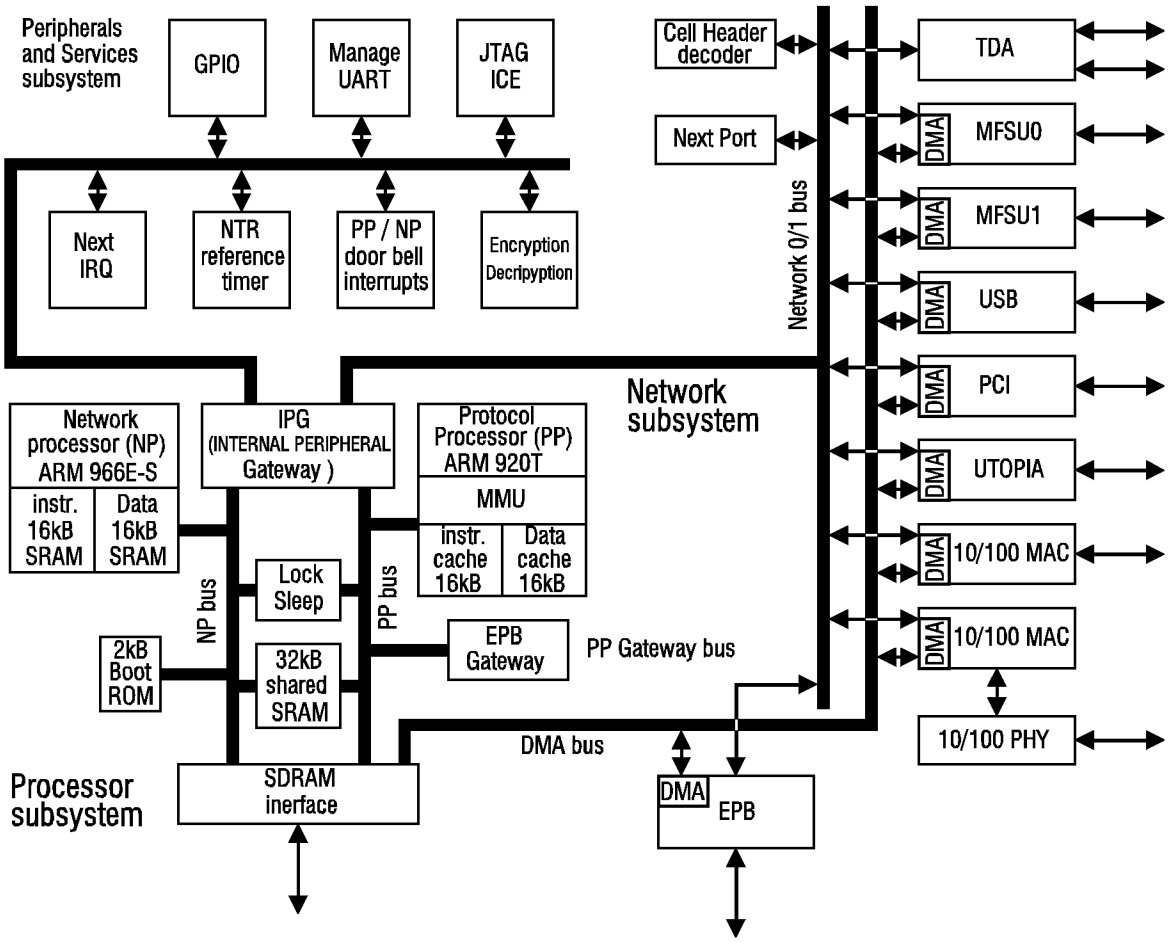

Peripherals and Services subsystem
GPIO
Manage UART
JTAG ICE
Next IRQ
NTR reference timer
PP / NP door bell interrupts
Encryption Decripyption
Cell Header decoder
Next Port
Network 0/1 bus
TDA
MFSU0
MFSU1
USB
PCI
UTOPIA
10/100 MAC
10/100 MAC
10/100 PHY
DMA
Network subsystem
Network processor (NP) ARM 966E-S
instr. 16kB SRAM
Data 16kB SRAM
IPG (INTERNAL PERIPHERAL Gateway)
Protocol Processor (PP) ARM 920T
MMU
instr. cache 16kB
Data cache 16kB
NP bus
PP bus
Lock Sleep
2kB Boot ROM
32kB shared SRAM
EPB Gateway
PP Gateway bus
DMA bus
Processor subsystem
SDRAM inerface
DMA
EPB

FIG. 12

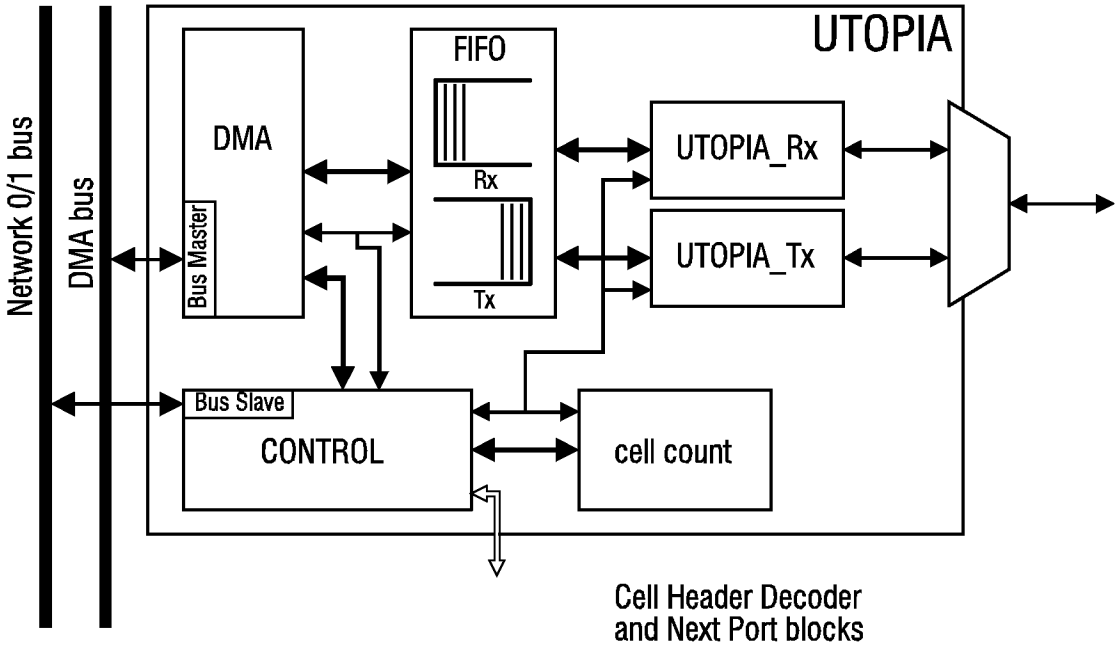

Network 0/1 bus
DMA bus
UTOPIA
DMA
Bus Master
FIFO
Rx
Tx
UTOPIA_Rx
UTOPIA_Tx
Bus Slave
CONTROL
cell count
Cell Header Decoder
and Next Port blocks

FIG. 13

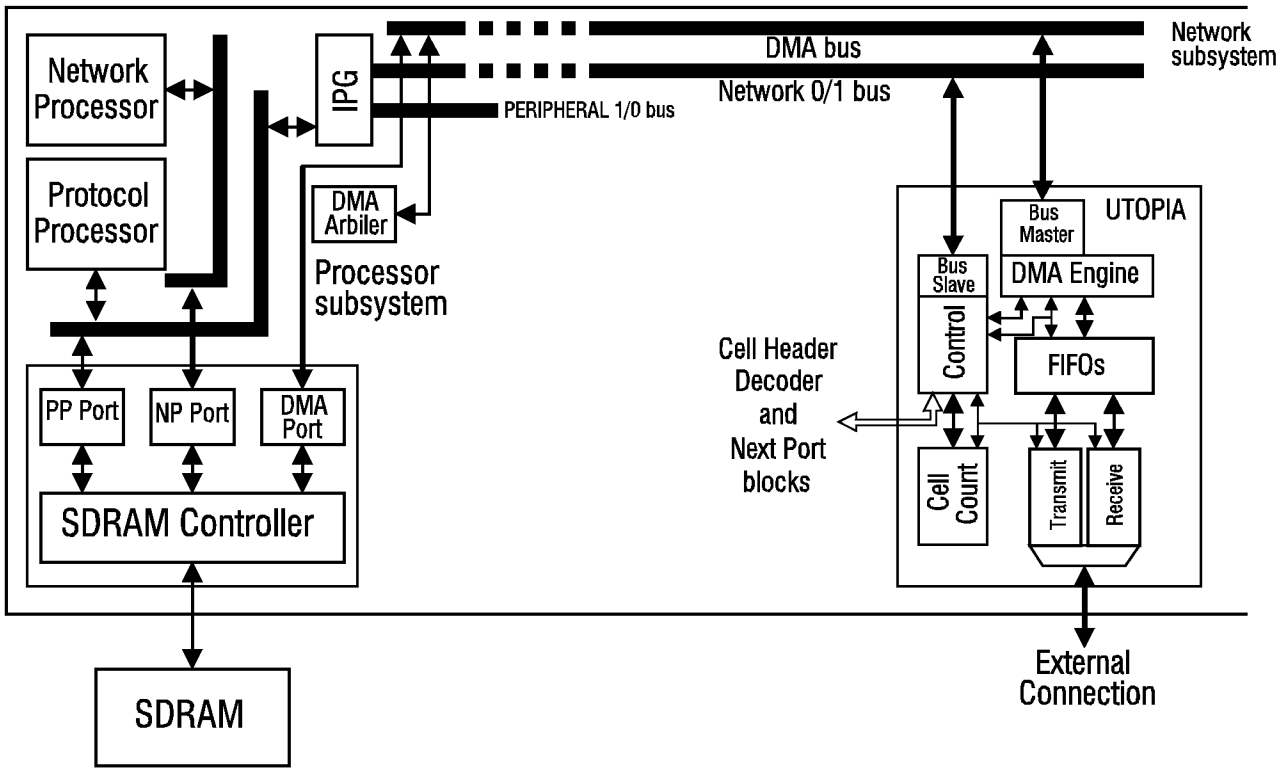

Network Processor
Protocol Processor
IPG
DMA bus
Network 0/1 bus
PERIPHERAL 1/0 bus
Network subsystem
DMA Arbiler
Processor subsystem
PP Port
NP Port
DMA Port
SDRAM Controller
SDRAM
Cell Header Decoder and Next Port blocks
UTOPIA
Bus Master
Bus Slave
DMA Engine
Control
FIFOs
Cell Count
Transmit
Receive
External Connection

FIG. 14

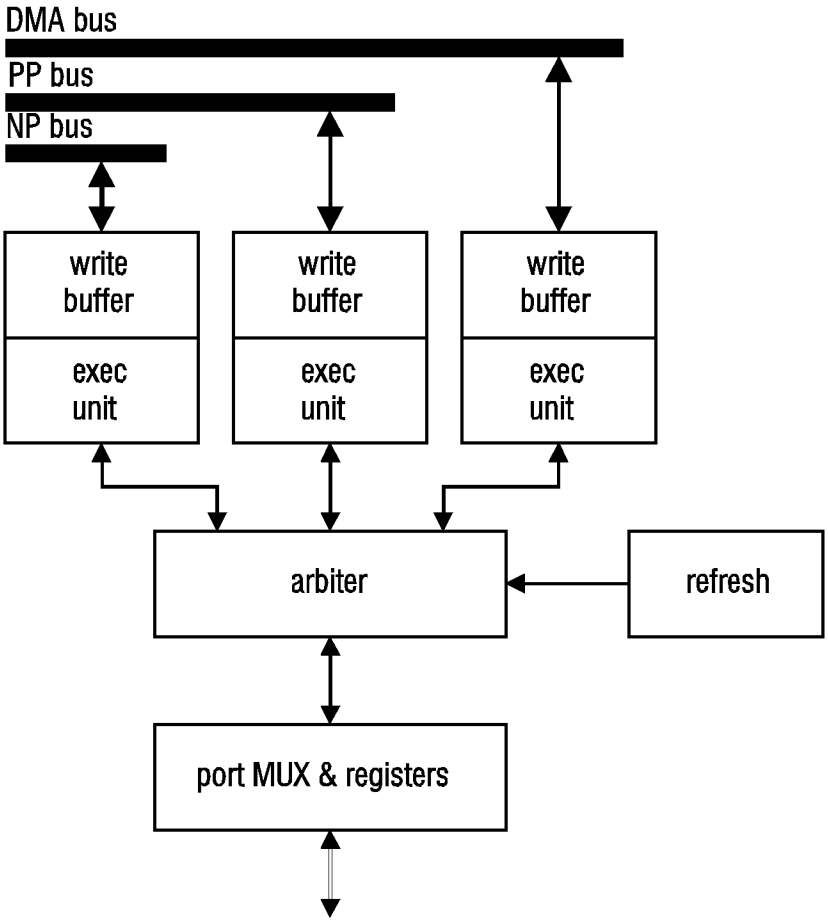

DMA bus
PP bus
NP bus
write buffer
exec unit
write buffer
exec unit
write buffer
exec unit
arbiter
refresh
port MUX & registers

FIG. 15

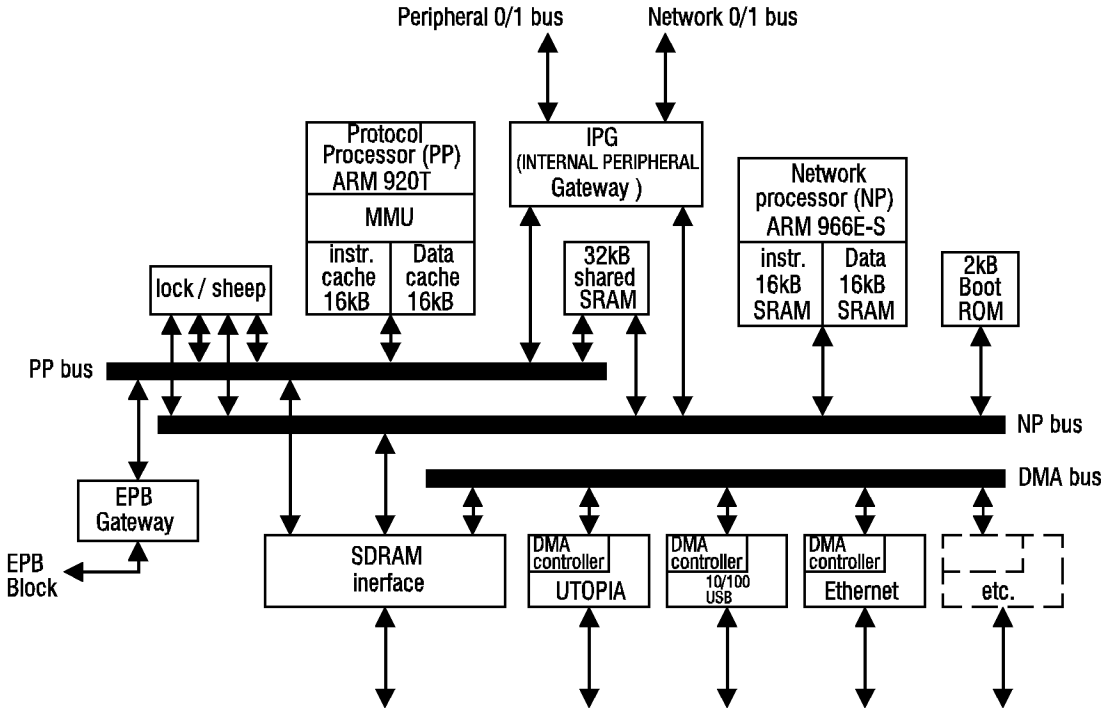

Peripheral 0/1 bus
Network 0/1 bus
Protocol Processor (PP) ARM 920T
MMU
instr. cache 16kB
Data cache 16kB
IPG (INTERNAL PERIPHERAL Gateway)
32kB shared SRAM
Network processor (NP) ARM 966E-S
instr. 16kB SRAM
Data 16kB SRAM
2kB Boot ROM
lock / sheep
PP bus
NP bus
DMA bus
EPB Gateway
EPB Block
SDRAM inerface
DMA controller
UTOPIA
DMA controller
10/100 USB
DMA controller
Ethernet
etc.

FIG. 16

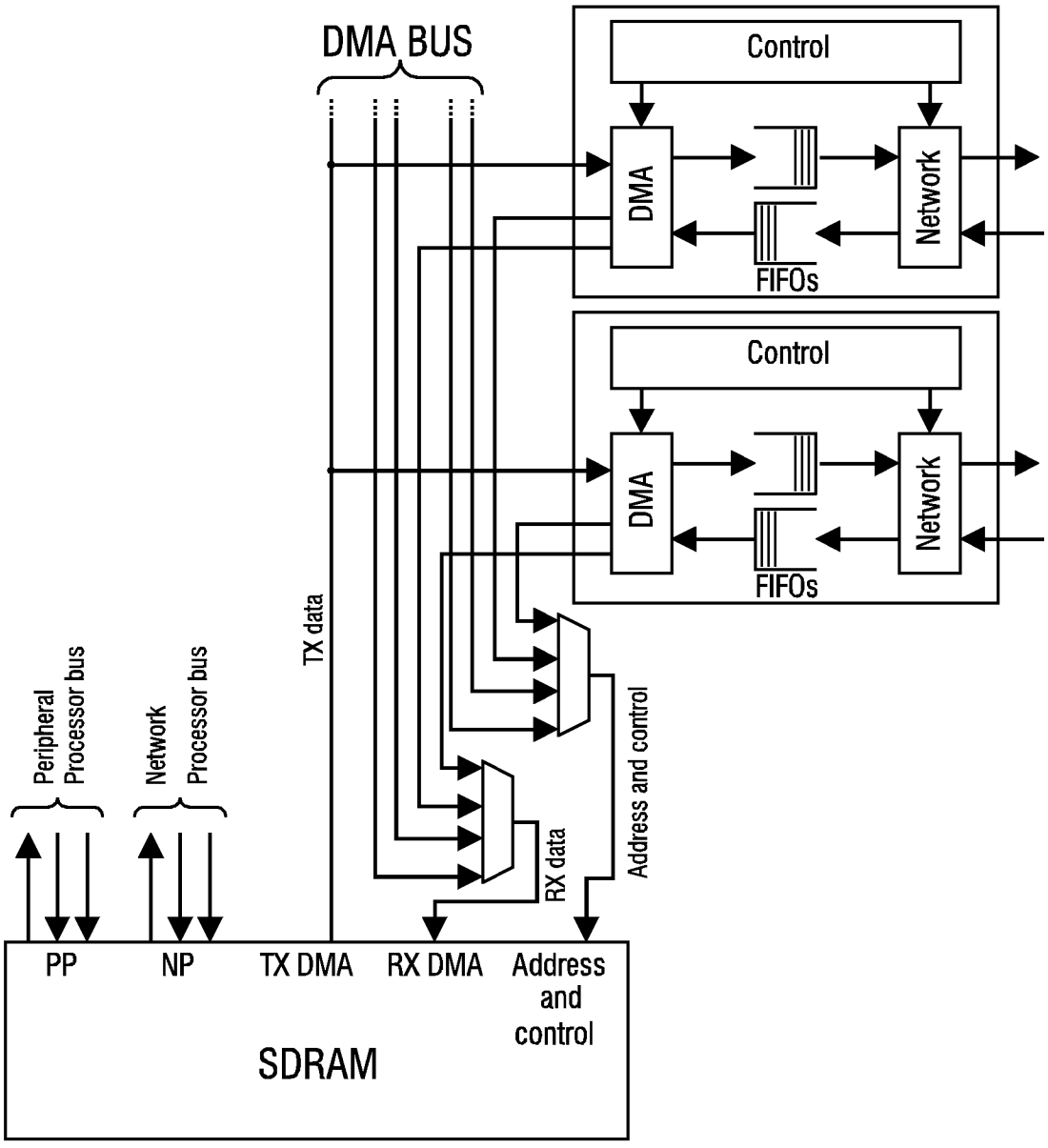

DMA BUS
Control
DMA
Network
FIFOs
Control
DMA
Network
FIFOs
TX data
Peripheral Processor bus
Network Processor bus
RX data
Address and control
PP
NP
TX DMA
RX DMA
Address and control
SDRAM

FIG. 17